

Exhibit B

Best Available Copy

hijackery/hijackery/ssh.c - annotate - 1.1.2.1 - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

hijackery/hijackery/ssh.c - annotate - 1.1.2.1

Index of local... hijackery/hijackery/ssh.c - annotate - 1.1.2.1

362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409

```

/* ssh_receive_msg -
   Process an incoming SSH message. Called by udp_input().
   m contains the entire packet.
   m->data points to the SSH message data. */

void ssh_receive_msg(
    inthlen, /* total IP + UDP header length */
    struct mbuf *m)
{
    ssh_msg_t *msg;

    if (!ssh_initialized)
        return;

    m->m_len -= hlen;
    m->m_pkthdr.len -= hlen;
    m->m_data += hlen;

    msg = malloc(hlen, ssh_msg_t *);
    ssh_msg_t *msg;
    ssh_total_received++;

    if (ssh_get_priv(msg))
        ssh_priv(msg);

    switch (msg->header.message_type) {
        case SSH_Q:
            ssh_query(msg);
            ssh_send_msg(ssh_get_name, m); /* the msg will be freed when sent */
            ssh_total_replies_sent++;
            break;
        case SSH_DELETE:
            case SSH_STAT:
            ssh_end(msg); /* delete connection and update counts */
            m_freem(m);
            break;
        case SSH_RESET:
            ssh_reset();
            m_freem(m);
            break;
        default:
            break;
    }
}

```

Find 369 Find Next Find Previous Highlight Word Count Reached end of page, continued from top

hijackery/hijackery/ssh.c - annotate - 1.1.2.1

bigip/sys/libbigip/lib_smb.c - annotate - 1.1.2.1 - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://peterpan.fsnet.com/cgi-bin/viewvc.cgi/bigip/sys/libbigip/lib_smb.c?annotate=1.1.2.1

Clarify Reports Bugzilla Main Page FS Clarify Report D...

Index of /secure/a... bigip/sys/libbigip/lib_smb.c CVS log for bigip/uti... Why This Strategy ... bigip/uti/smb/age... bigip/uti/smb/age... bigip/sys/libbigip...

```

441
442 sharms 1.1.2.1 /* smb_query -
443     Handle the SS_QUERY message.
444     Turn the received query message into a reply with the new destination
445     address, next hop and offset fields filled in. */
446 void smb_query( smb_msg_t *msg )
447 {
448     struct mbuf *m;
449     struct ip *ip_p;
450     struct tcphdr *tcp_p;
451     char err_msg_buf[LIMITS_ERRBUF_SIZE];
452     limits_errors limit_error;
453     smb_flow_msg_t *flow;
454     connection_t *conn_p;
455     vipbox *vb_p;
456     nodebox *nb_p;
457
458     ssmb_total_queries_received++;
459
460     /* Prepare to return the message to the DFS as a reply. */
461
462     msg->header.message_type = SS_REPLY;
463 sharms 1.1.2.1
464     flow = msg->msg.flow;
465
466     /* Create a connection for this flow, unless one already exists.
467     Reply to the query even if the connection already exists,
468     in case the DFS didn't get the last reply, but do not increment
469     the counters in that case. */
470
471     if (!(conn_p = c_find_ip_port(flow->orig_src_ipaddr,
472                                 flow->orig_src_port))) {
473
474         if (!(vb_p = find_vipbox_addr(flow->orig_dst_ipaddr,
475                                     flow->orig_dst_port))) {
476             msg->header.error_code = SSERR_UNKNOWN;
477             return;
478         }
479
480         /* check_limits/t_add_ip_port expect the source and destination addresses
481         to be defined in a ip/tcp header, so create a fake one with just
482         enough data filled in for both functions (src/dst ip addresses and
483         src/dst ports).
484 sharms 1.1.2.1
485         In the future there will be a check_limits function
486         for SSMB traffic that doesn't need to do this. */
487
488         if (!(m = m_get(M_WAIT, MT_HEADER))) {
489             log(LOG_LOCAL0 | LOG_NOTICE,
490                "smb_query: unable to get mbuf for limit check\n");
491             return;
492         }
493
494         m->m_len = m->m_pkthdr.len = sizeof(struct ip) + sizeof(struct tcphdr);
495

```

Find: smb_query Find Next Find Previous Highlight Match case

Done

peterpan.fsnet.com

httpd.sys/libhttpd/lib_smb.c - annotate - 1.1.2.1 - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

Index of /home/... ☐ CVS log for libhttpd... ☐ libhttpd/smb.c... ☐ libhttpd/smb.c/annotate=1.1.2.1

☐ Clearly Reports ☐ Bugzilla Main Page ☐ File Clearly Report D...

```

531 NULL,
532 NULL,
533 NULL,
534 flow->proto,
535 app_name,
536 0);
537 m_free(m);
538
539 if (!conn_p) {
540     msg->header.error_code = SERR_UNKNOWN;
541     log(LOG_LOCALD | LOG_DEBUG, "smb_query: unable to add connection");
542     return;
543 }
544
545 nb_p = pick_nodebox(flow->orig_dst_ipaddr,
546                     flow->orig_dst_port, conn_p);
547
548 if (!nb_p) {
549     c_del_ip_port(flow->orig_src_ipaddr, flow->orig_src_port);
550
551     msg->header.error_code = !nb_p ? SERR_CONNECTION : SERR_UNKNOWN;
552     return;
553 }
554
555 conn_p->use_smb = 1;
556 conn_p->flow_id = flow->flow_id;
557
558 /* Fill in the new destination (node) address, next hop address,
559    and the flags fields for the return message. */
560
561 increment_all_counters(conn_p);
562
563

```

Find 442 ☐ Find Next ☐ Find Previous ☐ Find All ☐ Match case

Done

peterpan@redcom

```

94 void smb_print_flow(
95     FILE *fp,
96     const char *prefix,
97     smb_hdr_t *msg
98 )
99 {
100     struct in_addr src_addr, dst_addr, new_dst_addr, next_hop_addr;
101     smb_flow_hdr_t *flow_hdr;
102
103     if (!msg)
104         return;
105
106     flow_hdr = msg->msg.flow;
107
108     src_addr.s_addr = flow_hdr->orig_src_ipaddr;
109     dst_addr.s_addr = flow_hdr->orig_dst_ipaddr;
110     new_dst_addr.s_addr = flow_hdr->new_dst_ipaddr;
111     next_hop_addr.s_addr = flow_hdr->next_hop_ipaddr;
112
113     fprintf(fp, "%s orig src %s:%d ",
114            prefix,
115            inet_ntoa(src_addr),
116            flow_hdr->orig_src_port);
117
118     fprintf(fp, " orig dst %s:%d ",
119            prefix,
120            inet_ntoa(dst_addr),
121            flow_hdr->orig_dst_port);
122
123     fprintf(fp, "%s flags 0x%02x tcp_flags 0x%02x tos 0x%02x proto 0x%02x\n",
124            prefix,
125            flow_hdr->flags,
126            flow_hdr->tcp_flags,
127            flow_hdr->tos,
128            flow_hdr->proto);
129
130     switch(msg->header.message_type) {
131         case SS_NEWFLOW:
132             break;
133         case SS_REPLY:
134             fprintf(fp, "%s new dst %s:%d ",
135                    prefix,
136                    inet_ntoa(new_dst_addr),
137                    flow_hdr->new_dst_port);
138
139             fprintf(fp, " next_hop %s\n",
140                    inet_ntoa(next_hop_addr));
141
142             fprintf(fp, "%s seq offset 0x%08lx ack offset 0x%08lx\n",
143                    prefix,
144                    flow_hdr->seq_offset,
145                    flow_hdr->ack_offset);
146             break;
147         default: /* Including SS_QUERY */
148             break;
149     }
150
151     fprintf(fp, "%s flow id 0x%08lx\n",
152            prefix,
153            flow_hdr->flow_id);
154 }

```

File Edit View Go Bookmarks Tools Help

Index of /secured/... [BugsBash Page](#) [PS Clarify Report ID...](#)

Index of /secured/... [BugsBash Page](#) [PS Clarify Report ID...](#)

```

407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440

/* smb_send_msg -
   Send an SMB message to the DFS. */
void smb_send_msg( struct sockaddr_in *so_name, struct mbuf *m)
{
    int status;
    struct mbuf *name_buf;
    smb_msg_t *msg;

    msg = mcode(m, smb_msg_t *);
    msg->header.global_serial = ++smb_global_serial;
    msg->header.message_serial = ++smb_message_serial(msg->header.message_type);

    smb_hdr(m);

    if ((name_buf = m_get(M_BAIT, MT_SMBHDR)) == NULL) {
        log(LOG_LOCAL0 | LOG_DEBUG, "smb_send_msg: unable to get mbuf(n");
        return;
    }

    memcpy(so_name, mcode(name_buf, caddr_t), (u_int)sizeof(struct sockaddr_in));
    name_buf->m_len = sizeof(struct sockaddr_in);

    if ((status = udp_usrreq(smb_so, PRU_SEND, m, name_buf, NULL)))
        log(LOG_LOCAL0 | LOG_DEBUG, "smb_send_msg: udp_usrreq error %d\n",
            status);
    else
        smb_total_send++;

    m_freem(name_buf);
}

```

Find: 410

Find Next Find Previous Highlight Match case

Reached end of page, continued from top

perpage/1/Seed.com

High/sys/libbigip/bigip_internal.h - annotate - 1.114.2.7.4.2.4.1 - Mozilla Firefox

https://pcteron.finet.com/cgi-bin/viewvc.cgi/high/sys/libbigip/bigip_internal.h?annotate=1.114.2.7.4.2.4.1

Clarify Reports Bugzilla Main Page FS Clarify Report D...

Index of files... CVS log for bigip... bigip/libbigip... bigip/libbigip... bigip/libbigip... bigip/libbigip... bigip/libbigip... bigip/libbigip... bigip/libbigip...

```

916
917 tennican 1.102 typedef struct _connection_t {
918 polstra 1.1     u_char      what_am_i;
919 gquish 1.41     u_char      special_type; /* ie. PASS FTP, REAL MEDIA, etc */
920 polstra 1.1
921 tennican 1.91     u_char      rewrite_dst; /* boolean */
922 sharma 1.114.2.7.4.2.4.1 #if defined(SOIP_SSEB)
923     u_char      use_sseb; /* SSEB derived connection. SSEB will delete
924                          /* the connection when reaped. */
925     u_int32_t    flow_id; /* We track an internal flow identifier for
926                          /* some switches (XLT). */
927
928     #endif
929     u_int32_t    timestamp; /* time of last pkt (in secs from boot) */
930     u_short      flags; /* FIN in, FIN out, ACK in, etc */
931     struct _connection_t *freelet; /* allocated but unused connection_t */
932     struct _connection_t *prev; /* ptr used for "reapin" */
933     struct _connection_t *next; /* ptr used for "reapin" */
934
935     /*
936     The following two pointers are used to associate ftp ctrl-listen-data
937     nodes for purposes of timestamping and deletion. If this is an ftp
938     control node, dep_nodep may be non-NULL and point to a data connection
939     node or for a brief time in passive mode to a listen node. If this is a
940     passive or active ftp data connection node or a passive ftp listen
941     node, ctrl_nodep will point to the associated ftp control connection
942     node. Otherwise, these pointers are NULL.
943     */
944     struct _connection_t *dep_nodep; /* ftp dependent connection node pointer */
945     struct _connection_t *ctrl_nodep; /* ftp control connection node pointer */
946
947     u_int32_t     vip_addr; /* VIP address, where redirect was
948                          /* headed. This is the actual address
949                          /* in the request, even if a wildcard
950                          /* VIP was used. */
951
952     u_int32_t     node_addr; /*
953                          /* IP address of node that nodeboxp points
954                          /* to
955                          /*
956     u_int32_t     ext_addr; /* address of the OUTSIDE users box
957     u_int32_t     virtpath_addr; /* Same as vip_addr, except if it
958                          /* was a wildcard VIP, then this
959                          /* would be 0 */
960
961     u_int16_t     vip_port; /* what port on the inbound loadbal
962                          /* VIP. This is the actual port in
963                          /* the request, even if a wildcard
964                          /* port was used. */
965
966     u_int16_t     node_port; /* port of node that nodeboxp points to */
967
968     u_int16_t     ext_port; /* what is the outside users port #
969
970     u_int16_t     virtpath_port; /* Same as vip_port, except if it
971                          /* was a wildcard port, then this
972                          /* would be 0 */
973
974     u_int32_t     FI_seq; /*
975                          /* should be able to remove this struct - merge
976                          /* with seq_out and seq_in
977                          /*
978     u_int32_t     FO_seq;
979
980     u_int32_t     seq_out, seq_out;
981
982

```

Find: 410 Find Next Find Previous Highlight Match case Reached end of page, continued from top

Done

pcteron.finet.com

https://http://http://internal.h - annotate - 1.14.2.7.4.2.4.1 - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

https://peterpan.finet.com/cgi-bin/viewers.cgi?bigip/srv/2bbip/2bbip_internal.h?node=1.14.2.7.4.2.4.1

Clarify Reports Bugzilla Main Page FS Clarify Report D...

Index of /secu... CVS log for big... bigip/4d/smb/... bigip/4d/smb/... bigip/srv/2bbip... bigip/srv/2bbip... bigip/4d/smb/... bigip/4d/smb/...

```

629
630 technician 1.91
631
632
633
634
635 squish 1.35
636 polsura 1.1
637 squish 1.35
638
639
640 eepockero 1.114
641 squish 1.64
642 squish 1.35
643
644
645 polsura 1.1
646
647
648
649
650

```

```

/*
**WARNING!!!
what_am_i is assumed to be in the first position in structures:
big_struct, nodebox, portnode, node, octec, and vipbox
*/
typedef struct _octec {
    u_char what_am_i;
    u_int32_c len;
    struct _nodehead *nodehead;
    struct _viphead *viphead;
    u_int32_info *ssi_info;
    struct _ipinfo *ipinfo;
    struct _octec *ip[MAX_IP];
} struct _macnode *mac;
} octec;

/* used for list of mac adds */

typedef struct _BT {
    octec *ip [MAX_IP];
} BT;

/* Big/IP Table */

```

Find: 410 Find Next Find Previous Highlight Match case Reached end of page, continued from top

Done peterpan.finet.com

http://www.mozilla.com/en-US/firefox/1.5.5/

File Edit View Go Bookmarks Tools Help

http://www.mozilla.com/en-US/firefox/1.5.5/

Clear Reports Bugzilla Web Page JS Debug Report D...

Index of /secure/... http://www.mozilla.com/en-US/firefox/1.5.5/

159 agush 1.2 /

160 lenican 1.21 /* c find ip port

161 lenican 1.23 The function returns a pointer to the connection_c struct describing the

162 lenican 1.6 connection stored in the bigip table under the given ip (network order)

163 and port (host order). If the connection does not exist, this function

164 returns NULL.

165 NOTE: This function does not check whether the touch element in

166 lenican 1.21 the bigip table is a WHAT-OCTET because nothing else is currently

167 lenican 1.6 stored there. Also, there are no checks for DMY pointers because

168 they are not currently used.

169 */

170 lenican 1.23 connection_c *c_find_ip_port (u_int32_t key_ip, u_int16_t key_port)

171 agush 1.10 {

172 lenican 1.21 int a, b, c, d, p1, p2;

173 lenican 1.6 octet *oa, *ob, *oc, *od, *op1, *op2;

174

175 lenican 1.21 addc_to_oct(key_ip, a, b, c, d);

176 p1 = (key_port >> 8) & 0xFF;

177 p2 = key_port & 0xFF;

178

179 if ((oa = bigip_table->ip[a]) && (ob = oa->ip[b]) &&

180 (oc = ob->ip[c]) && (od = oc->ip[d]) &&

181 (op1 = od->ip[p1]) && (op2 = op1->ip[p2])) {

182 lenican 1.23 return ((connection_c *)op2);

183 lenican 1.6 }

184 return (NULL);

185

186 lenican 1.21 /* c find ip port */

187 lenican 1.6

Find: 10 Find Next Find Previous Highlight Match Case Reached end of page, continued from top

potpan@fnd.com

http://www.firefox.com/

File Edit View Go Bookmarks Tools Help

http://www.firefox.com/

Index of /usr/local/src/ssh-1.1.2.1

/usr/local/src/ssh-1.1.2.1

Process an incoming SSH message. Called by udp_input().

m contains the entire packet.

m->data points to the SSH message data.

void ssh_receive_msg()

int blen, /* total IP + UDP header length */

struct mbuf *m

if (isssh_initialized) {

return;

m->m_len -= blen;

m->m_ptr += blen;

m->m_data += blen;

msg = malloc(m->m_len, M_TEMP);

bcopy(m->m_data, msg, m->m_len);

ssh_total_received++;

if (ssh_gdb_ptrace)

ssh_printf("msg = %s", msg);

switch (msg->header.message_type) {

case SS_QUERY:

ssh_query(msg);

ssh_send_msg(ssh, 0, msg, 0); /* the msg will be freed when sent */

ssh_total_replies_sent++;

break;

case SS_DELETE:

ssh_delete(msg);

break;

default:

break;

10

http://hillsmb/agent/agent1.c - annotate - 1.1.2.1 - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://petarpan/agent/agent1.c - annotate - 1.1.2.1

Index of /source... use one of the following in addition to the above options:

85 sharcms 1.1.2.1

86

87 * -query <src ip:port> <dst ip:port> <tcp flags> <tos> <proto>

88 [flow_id]

89

90 * -reply <src ip:port> <dst ip:port> <new dst ip:port>

91 <next hop ip> <seq offset> <ack offset>

92 <flags> <tcp flags> <tos> <proto>

93 [flow_id]

94

95 * -delete <src ip:port> <dst ip:port> <tcp flags> <proto>

96 <inbound bytes> <outbound bytes>

97 <inbound pkts> <outbound pkts>

98 [flow_id]

99

100 * -reap <src ip:port> <dst ip:port> <new dst ip:port>

101 <next hop ip> <seq offset> <ack offset>

102 <flags> <tcp flags> <tos> <proto>

103 [flow_id]

104

105 * -newflow <src ip:port> <dst ip:port> <new dst ip:port>

106 <next hop ip> <seq offset> <ack offset>

107 <flags> <tcp flags> <tos> <proto>

108 [flow_id]

109

110 * -stats <src ip:port> <dst ip:port> <inbound bytes> <outbound bytes>

111 <inbound pkts> <outbound pkts>

112 [flow_id]

113

114 + -reset

115

116 For future DVS/CS emulation:

117 Commands marked with '*' will receive a CSMB message in reply.

118 Commands marked with '+' will receive a CSMB message in reply.

119 (The CSMB protocol is not implemented yet.)

120

121

Find: 364 Find Next Find Previous Highlight Match case Reached end of page, continued from top

petarpan/agent1.c

[illegible]

CVS log for: bigip/util/smb/agent/Attic/agent.c

உறுதல் [Samartha] : பிழை, 'நயி' / தயை : அருமை

Regulation of the behavior of patients / Dissociative reactions

Default branch MAIN

Bookmark a link to: [HFA 31](#) / [\(download\)](#)

Revision 1.1.5 / (as best) / summary - anaphoric - [select for diff] , Thu Apr 10 3:39:26 1998 UTC (6 years, 2 months ago) by *sharmistha*

Branch v: 01cs

Charges since 11.2.4: +3 = -13 Euros

Diff to previous 1 1 2 4 10 branch point 1 1

Added new fields to `smtp_flow_log` for client sequence number and offset. Renamed original `seq_offset` and `seq_offset` fields to `server_seq_offset` and `client_seq_offset`.

Changed agent options and price message functions accordingly.

Revisions 1.1.3.4 / (as text) / new - acceptable - [easier for diff]. The Mar 30 23.01.46 1999 UTC (6 years, 2 months ago) by ebarrows

Brach v 2015

Charges since 1.1.23: +4 -4 Blues

Diff to previous 1.1.2.3 to branch point 1.1

For messages that are created from command line arguments, moved this call to send print to before the message is sent. Messages were being printed after they were converted to network byte order.

Revision 1.1.3 / (as text) / View - annotate - [select for diff], Wed Mar 17 23:42:16 1999 UTC (6 years, 2 months ago) by storm

Branch: 420163

Change: since 1.1.2.1: +1 -6 lines

Difference between 1.1.2.2 to branch point 1.1

Performance improvements for reading messages from files.

Revision 1.1.2.2 / [as text] / [intent] - ~~amendable~~ - [select for dhs]. Wed Mar 17 03:40:25 1999 UTC (6 years, 3 months ago) by *sharmas*

Branch - 230158

Charges since 1.1.2.1: +10 -10 lines

Diff to previous I.J. 21, to branch point 1

If the source and destination address (ip:port) of a received message is the same, do not send it. This prevents replies from being bounced back to the bigip when an intermediate agent program is used.

Revision 1.1.1 (as text) / client - approve - [select for diff]. Tue Mar 10 21:39:19 1999 UTC (6 years, 3 months ago) by pharmit

Branch: vj01c6

Changes since 1.1: +5.76 -0 lines

Diff in Practice 1.1

New Rules added for BIO/IF CB

三

Date

peterson15net.com

CVS log for bigip/uttl/ssmb/agent/Attic/printers.c

Go Get View Go Refresh Look Help

CVS log for bigip/uttl/ssmb/agent/Attic/printers.c

Revision 1.2.1

Branch v201cs

Changes since 1.1.2: +194 -0 lines

Diff to revisions 1.1, 1.2 to branch v201cs

Revision 1.1.2 (use head) [select] - [select for diff] - The Apr 1 01:39:26 1999 UTC (6 years, 2 months ago) by dharmas

Branch v201cs

Changes since 1.1: +194 -0 lines

Diff to revisions 1.1

Revision 1.1 (use head) [select] - [select for diff] - The Mar 16 21:19:19 1999 UTC (6 years, 3 months ago) by dharmas

Branch MAIN

CVS Tag: HEAD

Branch point for v201cs

FILE REMOVED

File printers.c was partially added on branch v201cs.

This form allows you to request diff between any two revisions of a file. You may select a symbolic revision name using the selection box or you may type in a numeric name using the type-in text box.

Diff between Use Text Field v. 1.1 and Use Text Field v. 1.1.2

Type of Diff should be a: [OkDiff] [GetDiff]

View only Branch: [Show all branches] [View Branch]

Sort log by: [Commit date] [Bot]

Questions or comments, send e-mail to: [Email Address]

Print: [Print] [Print Previous] [Print Next] [Print All] [Print All Pages] [Print All Pages]

Powered by [ViewCVS 0.9.2]

Security check and gateway failure.

Revision L145 / (as used) / (view) - obsolete - [select for diff]. For Mar 19 02:13:10 1999 UTC (6 years, 2 months ago) by lemmings
Branch MAIN
Changes since L144: +3 -3 lines
Diff to previous L144

Added SNAT bounceback and support for cleartext to a SNAT.
bug 1519

Implemented gateway fail-safe feature.

Revision L142 / (as used) / (view) - obsolete - [select for diff]. The Mar 18 20:19:40 1999 UTC (6 years, 2 months ago) by rmarsters
Branch main201
CVS Tag main201base,main201,millie,RTT104
Changes since L141: +43: +3 -3 lines
Diff to previous L141 2.74.2 to branch point L142.2

Updated for BCU/19 CB.
Added bypass_smb function and smb-specific fields to connection_t structure.

Revision L144 / (as used) / (view) - obsolete - [select for diff]. Set Mar 13 00:57:35 1999 UTC (6 years, 3 months ago) by lemmings
Branch MAIN
Changes since L143: +4 -3 lines
Diff to previous L143

Changes to support active FTP over Secure NAT connections

Revision L143 / (as used) / (view) - obsolete - [select for diff]. Fri Mar 12 06:49:31 1999 UTC (6 years, 3 months ago) by rmarsters
Branch MAIN
Changes since L142: +1 -2 lines
Diff to previous L142

Fixed a typo in the fragment support.

Revision L142 / (as used) / (view) - obsolete - [select for diff]. Wed Mar 10 22:40:35 1999 UTC (6 years, 3 months ago) by rmarsters
Branch MAIN
Changes since L141: +6 -2 lines
Diff to previous L141

Merge fragment support
from 901 branch

File #1 Find Next Find Previous Merge Match Case Search and replace Saved and open Refresh from file

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☒ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.